

CLAIMS

1. Method for updating software present in a first version (V1) in a device (40), by receiving and executing a delta file (D12) for upgrading said first version to a
5 second version (V2), which software is divided and stored in a number of memory blocks (201-204) defined in an address space (211) of a physical memory (48) in the device, comprising the steps of:
- defining an extra memory block (212) associated with said number of memory blocks, initially being placed before a first memory block located at one end of the
10 memory space;
 - erasing said extra memory block;
 - writing updated data for the first memory block on said extra memory block;
 - moving the extra memory block one block forward; and
 - processing all of said number of memory blocks one by one by
15 erasing said extra memory block,
writing updated data for the memory block adjacent to and after the extra memory block on the extra memory block,
moving said extra memory block forward one block at a time,
until all of said number of memory blocks have been shifted one step in the address
20 space.
2. The method as recited in claim 1, comprising the step of reversing the order of writing memory blocks in the address space, from said upgrade from a first version to a second version, to a second upgrade from said second version to a third version.
- 25
3. The method as recited in claim 2, comprising the steps of:
- said device, before receiving and executing a delta file, connecting with a server and communicating information related to a current address space status
determining which type of delta file is applicable for upgrading said software,
30 dependent on the present location of said extra block in said address space, wherein a first delta file type is adapted to upgrade the software present in said memory

blocks in one order in the address space, and a second delta file adapted to upgrade the software present in said memory blocks in a reversed order in the address space;

- downloading a delta file of the applicable delta file type from said server to said device; and

5 - upgrading said software using the applicable delta file.

4. The method as recited in any of the previous claims, comprising the step of modifying a start address within said address space in a boot code for said software, dependent on the moving of data between said memory blocks.

10

5. Computer program product, for use in a computer-controlled electronic device (40) for updating software present in a first version (V1) in the device, which device comprises means for receiving and executing a delta file (D12) for upgrading said first version to a second version (V2), which software is divided and stored in a

15 number of memory blocks (201-204) defined in an address space (211) of a physical memory (48) in the device, said computer program product comprising executable computer program code devised to cause the device to perform the steps of:

- defining an extra memory block (212) associated with said number of memory blocks, initially being placed before a first memory block located at one end of the

20 memory space;

- erasing said extra memory block;

- writing updated data for the first memory block on said extra memory block;

- moving the extra memory block one block forward; and

- processing all of said number of memory blocks one by one by

25 erasing said extra memory block,

writing updated data for the memory block adjacent to and after the extra memory block on the extra memory block,

moving said extra memory block forward one block at a time,

until all of said number of memory blocks have been shifted one step in the address

30 space.

6. The computer program product as recited in claim 5, comprising executable computer program code devised to cause the device to perform the steps of any of the previous claims 2-4.

5 7. Radio communication terminal (40) comprising a computer system (47) with associated computer code for updating software present in a first version (V1) in the terminal, and means (46) for receiving and executing a delta file (D12) for upgrading said first version to a second version (V2), which software is divided and stored in a number of memory blocks (201-204) defined in an address space (211)
10 of a physical memory (48) in the terminal, **characterised in** that an extra memory block (212) associated with said number of memory blocks is defined in said address space, initially being placed before a first memory block located at one end of the memory space, said terminal further comprising means for erasing said extra memory block; data writing means for writing updated data for the first memory
15 block on the extra memory block; means for moving the extra memory block one block forward; and means for processing all of said number of memory blocks one by one by
erasing said extra memory block,
writing updated data for the memory block adjacent to and after the extra
20 memory block on the extra memory block,
moving said extra memory block forward one block at a time,
until all of said number of memory blocks have been shifted one step in the address space.

25 8. The radio communication terminal as recited in claim 7, comprising means for reversing the order of writing memory blocks in the address space, from said upgrade from a first version to a second version, to a second upgrade from said second version to a third version.

30 9. The radio communication terminal as recited in claim 8, wherein said device is arranged to connect with a server before receiving and executing a delta file, for

communicating information related to a current address space status determining which type of delta file is applicable for upgrading said software, dependent on the present location of said extra block in said address space, wherein a first delta file type is adapted to upgrade the software present in said memory blocks in one order
5 in the address space, and a second delta file adapted to upgrade the software present in said memory blocks in a reversed order in the address space, said terminal comprising means for downloading a delta file of the applicable delta file type from said server to said device, and means for upgrading said software using the applicable delta file.

10

10. The radio communication terminal as recited in any of the previous claims 7-9, comprising means for modifying a start address within said address space in a boot code for said software, dependent on the moving of data between said memory blocks during writing as determined by said delta file.

15

CLAIMS

1. Method for updating software present in a first version (V1) in a device (40), by receiving and executing a delta file (D12) for upgrading said first version to a
5 second version (V2), which software is divided and stored in a number of memory blocks (201-204) defined in an address space (211) of a physical memory (48) in the device, comprising the steps of:

- defining an extra memory block (212) associated with said number of memory
10 blocks, initially being placed before a first memory block located at one end of the memory space;
- ~~erasing said extra memory block to define an extra memory block space;~~
- writing updated data for ~~the~~ first of said number of memory blocks, as ~~determined by said delta file,~~ on said extra memory block;
- 15 - ~~erasing said first memory block, thereby moving said~~ the extra memory block space one block forward; and
- processing all of said number of memory blocks one by one by
erasing said extra memory block,
writing updated data for the memory block adjacent to and after the extra
20 memory block on the extra memory block,
moving said extra memory block forward one block at a time,
until all of said number of memory blocks have been shifted one step in the address space. ~~writing updated data for a second of said number of memory blocks, as determined by said delta file, on said first memory block.~~

25

2. ~~The method as recited in claim 1, wherein said extra memory block is defined as a memory block placed adjacent to said first memory block in said address space.~~

3.1. ~~The method as recited in claim 1, comprising the step of processing all of said~~
30 ~~number of memory blocks one by one by moving said extra memory block space one block at a time; and writing data on that of said number of memory blocks on~~

~~which an extra memory block space is defined, from the next memory block in said address space, during upgrading as determined by said delta file, until all of said number of memory blocks have been shifted one step in the address space.~~

- 5 4.2. The method as recited in claim 1, comprising the step of reversing the order of writing memory blocks in the address space, from said upgrade from a first version to a second version, to a second upgrade from said second version to a third version.

5.3. The method as recited in claim 24, comprising the steps of:

- 10 - said device, before receiving and executing a delta file, connecting with a server and communicating information related to a current address space status determining which type of delta file is applicable for upgrading said software, dependent on the present location of said extra block in said address space, wherein a first delta file type is adapted to upgrade the software present in said memory
- 15 blocks in one order in the address space, and a second delta file adapted to upgrade the software present in said memory blocks in a reversed order in the address space;
- downloading a delta file of the applicable delta file type from said server to said device; and
- upgrading said software using the applicable delta file.

20

6.4. The method as recited in any of the previous claims, comprising the step of modifying a start address within said address space in a boot code for said software, dependent on the moving of data between said memory blocks ~~during writing as determined by said delta file.~~

25

- 7.5. Computer program product, for use in a computer-controlled electronic device (40) for updating software present in a first version (V1) in the device, which device comprises means for receiving and executing a delta file (D12) for upgrading said first version to a second version (V2), which software is divided and stored in a
- 30 number of memory blocks (201-204) defined in an address space (211) of a physical memory (48) in the device, said computer program product comprising executable

computer program code devised to cause the device to perform the steps of:

- defining an extra memory block (212) associated with said number of memory blocks, initially being placed before a first memory block located at one end of the memory space;

5 - erasing said extra memory block;

- writing updated data for the first memory block on said extra memory block;

- moving the extra memory block one block forward; and

- processing all of said number of memory blocks one by one by
erasing said extra memory block,

10 writing updated data for the memory block adjacent to and after the extra
memory block on the extra memory block,

moving said extra memory block forward one block at a time,

until all of said number of memory blocks have been shifted one step in the address
space.

15 ~~- defining an extra memory block with said number of memory blocks;~~

~~- erasing said extra memory block to define an extra memory block space;~~

~~- writing updated data for a first of said number of memory blocks, as determined~~
~~by said delta file, on said extra memory block;~~

~~- erasing said first memory block, thereby moving said extra memory block space~~

20 ~~one block; and~~

~~- writing updated data for a second of said number of memory blocks, as determined~~
~~by said delta file, on said first memory block.~~

8.6. The computer program product as recited in claim 57, comprising executable
25 computer program code devised to cause the device to perform the steps of any of
the previous claims 24-64.

9.7. Radio communication terminal (40) comprising a computer system (47) with
associated computer code for updating software present in a first version (V1) in the
30 terminal, and means (46) for receiving and executing a delta file (D12) for
upgrading said first version to a second version (V2), which software is divided and

stored in a number of memory blocks (201-204) defined in an address space (211) of a physical memory (48) in the terminal, **characterised in** that an extra memory block (212) associated with said number of memory blocks is defined in said address space, initially being placed before a first memory block located at one end

5 of the memory space, said terminal further comprising means for erasing said extra memory block ~~for defining an extra memory block space~~; data writing means for writing updated data for ~~the~~ a first of said number of memory blocks, as determined by said delta file, on said ~~the~~ extra memory block; means for moving the extra memory block one block forward; and means for processing all of said number of

10 memory blocks one by one by
erasing said extra memory block,
writing updated data for the memory block adjacent to and after the extra memory block on the extra memory block,
moving said extra memory block forward one block at a time,

15 until all of said number of memory blocks have been shifted one step in the address space. ~~data erasing means for erasing said first memory block for moving said extra memory block space one block; wherein said data writing means are devised to write updated data for a second of said number of memory blocks, as determined by said delta file, on said first memory block.~~

20

~~9. The radio communication terminal as recited in claim 9, wherein said extra memory block is placed adjacent to said first memory block in said address space.~~

~~10. The radio communication terminal as recited in claim 9, wherein said data~~

25 ~~writing means are devised to process all of said number of memory blocks one by one from said extra memory block by writing data on one of said number of memory blocks from the next memory block in said address space, during upgrading as determined by said delta file, until all of said number of memory blocks have been shifted one step in the address space.~~

30

12.8. The radio communication terminal as recited in claim 79, comprising means

for reversing the order of writing memory blocks in the address space, from said upgrade from a first version to a second version, to a second upgrade from said second version to a third version.

- 5 ~~13~~.9. The radio communication terminal as recited in claim ~~8~~12, wherein said device is arranged to connect with a server before receiving and executing a delta file, for communicating information related to a current address space status determining which type of delta file is applicable for upgrading said software, dependent on the present location of said extra block in said address space, wherein
- 10 a first delta file type is adapted to upgrade the software present in said memory blocks in one order in the address space, and a second delta file adapted to upgrade the software present in said memory blocks in a reversed order in the address space, said terminal comprising means for downloading a delta file of the applicable delta file type from said server to said device, and means for upgrading said software
- 15 using the applicable delta file.

- ~~14~~.10. The radio communication terminal as recited in any of the previous claims ~~97-913~~, comprising means for modifying a start address within said address space in a boot code for said software, dependent on the moving of data between said memory blocks
- 20 during writing as determined by said delta file.